Atty. Docket No. MS150905.1

SYSTEM, REPRESENTATION, AND METHOD PROVIDING MULTILEVEL INFORMATION RETRIEVAL WITH CLARIFICATION DIALOG

by

Eric J. Horvitz and Susan T. Dumais

CERTIFICATION

I hereby certify that the attached patent application (along with any other paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on this date June 28, 2001, in an envelope as "Express Mail Post Office to Addressee" Mailing Label Number EL798606666US addressed to the: Box Patent Application, Assistant Commissioner for Patents, Washington, D.C. 20231.

Himanshu S. Amin
(Typed or Printed Name of Person Mailing Paper)

(Signature of Person Mailing Paper)

Title: SYSTEM, REPRESENTATION, AND METHOD PROVIDING MULTILEVEL INFORMATION RETRIEVAL WITH CLARIFICATION DIALOG

Technical Field

The present invention relates generally to computer systems, and more particularly to a system and method to improve information search and retrieval utilizing a multilevel analysis wherein users are guided to desired information based upon determined probabilities and feedback received from a clarification dialog.

Background of the Invention

Search information and retrieval systems are common tools enabling users to find desired information relating to a topic. From Web search engines to desktop application utilities (*e.g.*, help systems), users consistently utilize information and retrieval systems to discover unknown information about topics of interest. In some cases, these topics are prearranged into topic and subtopic areas. For example, "Yahoo" provides a hierarchically arranged predetermined list of possible topics (*e.g.*, business, government, science, *etc.*) wherein the user will select a topic and then further choose a subtopic within the list. Another example of predetermined lists of topics is common on desktop personal computer help utilities wherein a list of help topics and related subtopics are provided to the user. While these predetermined hierarchies may be useful in some contexts, users often need to search for/inquire about information outside of and/or not included within these predetermined lists. Thus, search engines or other search systems are often employed to enable users to direct user-crafted queries in order to find desired information. Unfortunately, this often leads to frustration when many unrelated files are retrieved since users may be unsure of how to author or craft a particular query. This often causes users to continually modify queries in order to refine retrieved search results to a reasonable number of files.

As an example of this dilemma, it is not uncommon to type in a word or phrase in

a search system input query field and retrieve several thousand files - or millions of web sites in the case of the Internet, as potential candidates. In order to make sense of the large volume of retrieved candidates, the user will often experiment with other word combinations to further narrow the list since many of the retrieved results may share common elements, terms or phrases yet have little or no contextual similarity in subject matter. This approach is inaccurate and time consuming for both the user and the system performing the search. Inaccuracy is illustrated in the retrieval of thousands if not millions of unrelated files/sites the user is not interested in. Time and system processing speed are also sacrificed when searching massive databases for possible yet unrelated files.

Generally, conventional search systems will search for information in a flat or non-hierarchical manner that can exacerbate the accuracy and speed problems described above. In other words, the search system will attempt to match or find all topics relating to the user's query input regardless of whether the "searched for" topics have any contextual relationship to the topical area or category of what the user is actually interested in. As an example, if a user were to input the word "Saturn" into a conventional search system, all types of unrelated results are likely to be "searched for" and returned such as relating to cars, car dealers, planets, computer games, and other sites having the word "Saturn". Similar results are also achieved with phrases wherein many unrelated topics may be "searched for" and returned in response to the combination of words in the phrases. Consequently, as a result of conventional flat, non-hierarchical search architectures, wherein substantially all potential topics are analyzed for common elements or element combinations of the query input, and as a result of ever increasing databases for storing these topics, user frustrations continue to grow and system performance decreases as more and more potential "search results" are returned having related elements but little contextual similarity.

In view of the above problems associated with conventional search and information retrieval systems, there is a need for a system and/or methodology to mitigate search and retrieval of unrelated information and to facilitate finding information without having to continually author or craft ever more sophisticated queries.

Summary of the Invention

The following presents a simplified summary of the invention in order to provide a basic understanding of some aspects of the invention. This summary is not an extensive overview of the invention. It is intended to neither identify key or critical elements of the invention nor delineate the scope of the invention. Its sole purpose is to present some concepts of the invention in a simplified form as a prelude to the more detailed description that is presented later.

The present invention relates to a system and methodology to facilitate improved search and retrieval of information. Information is analyzed according to a hierarchical architecture of L layers, wherein respective layers are further analyzed according to N broader categorical areas, the categorical areas having one or more possible topics, wherein L and N are integers. It is noted that categories, topics and/or subtopics are distinguished as such for exemplary purposes, however, it is to be appreciated that similar processing (*e.g.*, algorithms) can be employed across/within these respective designations. The categorical areas and topics can have automatically determined and/or assigned probabilities (*e.g.*, weights). These probabilities are based upon a determination regarding the likelihood that information related thereto will be associated with a query (*e.g.*, user and/or system input requesting information) that is directed at seeking information relating to the areas and/or topics. The probabilities may be determined automatically *via* classifiers that are hierarchically configured (*e.g.*, higher level classifiers determining sublevel classifiers) and are constructed to analyze the areas and related topics. The hierarchical analytical approach facilitates both speed and accuracy over conventional systems. High-level classifiers enable search requests to be narrowed to a more definite categorical area in a rapid manner, wherein sub-classifiers employed for each categorical area may be trained to provide more accurate models for retrieving desired information within a category. These classifiers may include hierarchically configured Support Vector Machines, Naive Bayes, Bayes Net, decision tree learning models, similarity-based and/or other vector-based information retrieval methods such as cosine-based computation, for example. A hand-crafted approach may also be applied to assign predetermined probabilities to a hierarchical analytical structure of areas and topics.

A dialog (*e.g.*, audio feedback, text feedback) is employed in order to aid the user in refining and distinguishing between one or more possible areas the user may be interested in - but, possibly unsure of how to inquire about. For example, a user may direct a query in accordance with the present invention relating to a particular information topic (*e.g.*, formatting). As an example, the dialog may initially ask the user to disambiguate between formatting text, formatting hard drives, or formatting printers based upon determined probabilities relating to the query and the categorical areas. If the user selects text as a broad area, for example, further dialog may be initiated wherein the next most probable areas relating to text formatting are presented. In this manner, a user may retrieve information without having to understand which word or phrase combinations are necessary to acquire information on a given subject. Moreover, based upon the user's response to the dialog, the user is presented with the most likely area of interest in subsequent dialog feedback to further refine the search and without having to peruse unrelated information. Dialog ranges from sophisticated text-generation engaging the user to clarify the area of focus, to a simpler presentation of a list containing topics that the user may desire to review, coupled with a question seeking feedback from the user that can disambiguate the focus of the list.

It is to be appreciated that substantially any type of information may be similarly analyzed and retrieved. For example, web and/or other information may be hierarchically analyzed according to automatically determined areas and topics wherein the dialog may be employed to further aid the user in retrieving related topic information. In accordance with another aspect of the present invention, data logs and user activity monitoring may be employed to further refine classification models and/or include other categories or topics.

According to one aspect of the present invention, an information retrieval system is provided. The system includes a hierarchal analysis component that receives a query and processes probabilities associated with N categories, each category having one or more topics, N being an integer, and an interactive component that provides feedback derived from the query and the probabilities associated with the N categories and the one or more topics. The feedback is utilized to determine at least one category of the N categories to facilitate retrieval of at least one of the one or more topics.

The following description and the annexed drawings set forth in detail certain illustrative aspects of the invention. These aspects are indicative, however, of but a few of the various ways in which the principles of the invention may be employed and the present invention is intended to include all such aspects and their equivalents. Other advantages and novel features of the invention will become apparent from the following detailed description of the invention when considered in conjunction with the drawings.

Brief Description of the Drawings

Fig. 1 is a schematic block diagram illustrating a hierarchical information and retrieval system in accordance with an aspect of the present invention;

Fig. 2 is a schematic block diagram illustrating an automatic classifier construction system and an automatic classification system in accordance with an aspect of the present invention;

Fig. 3 is a schematic block diagram illustrating a more detailed automatic classification system in accordance with an aspect of the present invention;

Fig. 4 is a schematic block diagram illustrating a more detailed information and retrieval system in accordance with an aspect of the present invention;

Fig. 5 is a schematic block diagram illustrating a central server and data store in accordance with an aspect of the present invention;

Fig. 6 is a diagram illustrating a top-level display presentation in accordance with an aspect of the present invention;

Fig. 7 is a diagram illustrating a sub-level display presentation in accordance with an aspect of the present invention;

Fig. 8 is a diagram illustrating an alternative hierarchical model in accordance with an aspect of the present invention;

Fig. 9 is a flow diagram illustrating a methodology providing improved information search and retrieval in accordance with an aspect of the present invention; and

Fig. 10 is a schematic block diagram illustrating a suitable computing environment in accordance with an aspect of the present invention.

Detailed Description of the Invention

The present invention relates to a system and methodology to facilitate improved search and retrieval of information. A modular and coherent analytical system and process is provided wherein large amounts of data are initially analyzed according to a hierarchical determination of broad categorical areas of interest associated with a query. Respective categorical areas are further related to a plurality of topics and subtopics of information pertaining to that area. Dialog and/or other types of feedback are employed to enable disambiguation amongst the categorical areas. Based upon the feedback and the query, a subsequent search for desired information is conducted in a more narrow and precise range of topics associated with one or more of the categorical areas. In this manner, accuracy and speed is improved over conventional search and information systems that attempt to retrieve information from all possible topics associated with the query in a flat or non-hierarchal manner. Accuracy is improved since the feedback provided by the user enables refinement of the search to topics that are most likely of interest to the user – mitigating having the user experiment with alternate query formulations to find desired information. Moreover, the user is less likely to have to peruse a list of retrieved information files that may have common elements but no subject matter relationship (*e.g.*, "Mustang" search retrieving files relating to cars, horses, music). Retrieval speed and performance is improved since the search is narrowed to a categorical area. Thus, unrelated topics are substantially excluded from the refined search of the present invention thereby improving system performance. In addition, more accurate classifiers can be developed to distinguish among items within a categorical area.

As used in this application, the term "component" is intended to refer to a computer-related entity, either hardware, a combination of hardware and software, software, or software in execution. For example, a component may be, but is not limited to, a process running on a processor, a processor, an object, an executable, a thread of execution, a program, and a computer. By way of illustration, both an application running on a server and the server can be a component.

Referring initially to Fig. 1, a system 10 illustrates a hierarchical information search and retrieval system in accordance with an aspect of the present invention. The

system 10 includes a hierarchically determined architecture of probabilistic layers 20, a hierarchy analysis component 30 and a query/dialog interface 34. The layers 20 can include a meta layer 38 for determining probabilities P of a category layer 40. The category layer 40 includes several broader category areas that are denoted as C1 through CN, N being an integer, wherein each category area is associated with a plurality of topics denoted as TK1 through TKE, TL1 through TLF, and TM1 through TMG, wherein E, F, and G are integers, respectively. It is to be appreciated that although three layers are depicted in the layers 20, other layers and hierarchies are possible. As will be described in more detail below, the probabilities P may be automatically determined by hierarchically configured classifiers wherein a top-level classifier is configured for the meta layer 38 and sublevel classifiers are configured for each category area C1-CN. It is noted that probabilities may also be predetermined for the layers 20 and will be described in more detail below according to an alternative aspect of the present invention in relation to Fig. 8.

The topics may refer to substantially any information or subject matter. The categories provide a broader classification of the topics that relate to a particular category. For example, the categories may refer to well-known areas of a help utility such as printing, formatting, creating, viewing, editing, saving, *etc.* wherein the topics are then related to each category. As an example, there may be a "properties" topic associated with each category that provides different information on properties depending on which category the topic is associated. It is to be appreciated that substantially any type of information may be similarly analyzed and classified in accordance with the present invention. This information may include Web content and/or other remote/local data sources wherein topical information is stored.

The query/dialog interface 34 receives a query input 42 that is subsequently processed by the hierarchy analysis component 30 (HAC). This processing includes determining the most likely category areas to be displayed to the user and also may subsequently include retrieved results of likely topics associated with a category. The HAC 30 receives the query input and analyzes the input in conjunction with the meta layer 38. The meta layer 38 provides probabilities of the most likely categories associated with the query input 42 to the HAC 30. Based upon the probabilities and subsequent

analysis that is described in more detail below, the HAC 30 may provide feedback 44 to the query dialog interface 34 to inquire which of the categories the user is most likely interested in at a display output 46 and/or other type of computer system output. It is noted that users may select one or more categories of interest. Based upon a selection of a desired category and/or categories by the user at the interface 34, the HAC 30 then drives the category layer 40 along with the query input to retrieve related topics associated with the selected category. These topics may then be provided to the query/dialog interface 34 wherein the results may be displayed to the user at the output 46.

By providing the hierarchical architecture of the present invention, a modular and coherent structure is formed thus enabling developers to modularize tasks within categories and topics. Thus, tedious and time-consuming relationships do not have to be predetermined between all topics and categories. In this manner, a development team can be assigned to each category with responsibilities for providing only those topics pertaining to that category. By utilizing clarification from the user regarding the desired category of interest, the present invention improves system speed and accuracy over conventional search systems. Speed is improved because the clarification feedback enables the system 10 to refine searches to the category of interest without searching other categories and topics. Accuracy is improved since more accurate sublevel classifiers may be constructed for the topics when the sublevel classifiers are refined to a narrower and more well-defined area.

Referring now to Fig. 2, a system 50 illustrates an automatic classifier construction system 54 (ACCS) for building run-time classifiers within an automatic classification system 58 (ACS). The ACS 58 determines the probabilities described above in relation to Fig. 1 for the layers 20. A top-level classifier 60 is configured by the ACCS 54 to determine probabilities for the meta layer 38. In response to a query input vector, the top-level classifier 60 determines probabilities of the most likely category areas associated with the query input. Based on the probabilities determined by the top-level classifier 60, the query input, and the clarification feedback described above, sublevel classifiers 1 through C, C being and integer, associated with each category are then employed to determine probabilities of the most likely topics within the category

area. In this manner, a hierarchy of classifiers is provided wherein a query is first refined to a possible area of interest *via* the top-level classifier 60 and then utilized with a more narrowly defined sublevel classifier 1 through C for each category area in order to retrieve desired topic results that are contextually related.

According to one aspect of the invention Support Vector Machines (SVM) which are well understood are employed as the classifiers. It is to be appreciated that other classifier models may also be employed such as Naive Bayes, Bayes Net, decision tree, similarity-based, vector-based, and/or other learning models. SVM's are configured *via* a learning or training phase within the ACCS 54. A classifier is a function that maps an input attribute vector, $\mathbf{x} = (x1, x2, x3, x4, xn)$, to the confidence that the input belongs to a class—that is, $f(\mathbf{x}) = confidence(class)$. In the case of topic classification, attributes are words in a query or other domain-specific attributes derived from the words in a query (*e.g.,* parts of speech, presence of key terms), and the classes are the categories or areas of interest. An important aspect of SVMs and other inductive-learning approaches is to employ a training set of labeled instances to learn a classification function automatically. The training set is depicted within a data store 53 associated with the ACCS 54. As illustrated, the training set may include a subset of queries 1 through I that indicate potential and/or actual elements or element combinations (*e.g.,* words or phrases) that are employed to inquire about a particular topic. As illustrated, the data store 53 also includes a plurality of topics 1 through T. Each query can be associated with one or more topics (*e.g.,* (Q1,T2,T3,T9), (Q7,T2,T6), (Q2,T5)). During learning, a function that maps the input features to a confidence of class is learned. Thus, after learning a model, respective topics are represented as a weighted vector of input features. It is noted that other implementations of queries and/or topics are possible. For example, another generalization can be employed to train not only on queries for topics and subtopics, but also on the raw text associated with a target content and/or documents. In other words, a system can be booted with a few queries, but provided with a plurality of raw text, and also add not only queries but raw text later to enhance the system.

For topic classification, binary feature values (*e.g.,* a word occurs or does not occur in a topic), or real-valued features (*e.g.,* a word occurs with importance weight r) are often employed. Since topic collections may contain a large number of unique terms,

a feature selection is generally employed when applying machine-learning techniques to topic categorization. To reduce the number of features, features may be removed based on overall frequency counts, and then selected according to a smaller number of features based on a fit to the categories. The fit to category can be determined *via* mutual information, information gain, chi-square and/or any other statistical selection techniques. These smaller descriptions then serve as input to the SVM. It is noted that linear SVMs provide suitable generalization accuracy and provide fast learning. Other classes of nonlinear SVMs include polynomial classifiers and radial basis functions and may also be utilized with the present invention.

The ACCS 54 employs a learning model 55 in order to analyze the queries and topics in the data store 53 to learn a function mapping input vectors to confidence of class. For many learning models, including the SVM, the model for each category can be represented as a vector of feature weights, **w** 56 (*e.g.*, w1, w2,..wv). Thus, there is a learned vector of weights for each category. When the weights 56 are learned, new queries are classified by computing the dot product of **x** and **w**, wherein **w** is the vector of learned weights for the respective categories, and **x** is the vector representing a new query. A sigmoid function may also be provided to transform the output of the SVM to probabilities. Probabilities provide comparable scores across categories or classes.

An SVM is a parameterized function whose functional form is defined before training. Training an SVM generally requires a labeled training set, since the SVM will fit the function from a set of examples. The training set consists of a set of E examples, E being an integer. Each example consists of an input vector, $\boldsymbol{x}$, and a category label, y, which describes whether the input vector is in a category. For each category there are E free parameters in an SVM trained with E examples. To find these parameters, a quadratic programming (QP) problem is solved as is well understood. There is a plurality of well-known techniques for solving the QP problem. These techniques may include a Sequential Minimal Optimization technique as well as other techniques such as chunking.

Turning now to Fig. 3, a more detailed illustration is provided for the ACS 58. A query input 70 that has been transformed into an input vector **x** is applied to the top-level classifier 60 for each category. The top-level classifier 60 utilizes the learned weight vectors **w** determined by the ACCS 54 (one weight vector for each top-level category)

and forms a dot product to provide an output 72 of category probabilities P. As will be described in more detail below in relation to Fig. 4, the category probabilities are analyzed by an analytical component to provide clarification and/or dialog feedback to a user. The feedback provides a display of the most likely categories that are related to the user's query. Based upon some input from the user clarifying/selecting one or more categories of interest C1 through CN, sublevel classifiers are then employed that relate to the category areas selected by the user. The particular sublevel classifier then utilizes the query input to determine probabilities for topics that are associated with the category areas. The analytical component may then display the relevant topics based upon the determined probabilities. In this manner, a hierarchical system is formed wherein category areas are first disambiguated by enabling the user to select the most relevant contextual category area. The sublevel classifiers may then be constructed according to the confines of the particular category thus enabling much more accurate searches within the topics. Furthermore, system speed is improved by narrowing the search to a category area of contextually related topics since all other category areas may be excluded during the search. It is noted that a single weight vector is depicted in the top-level classifier 60 and the sublevel classifiers 1 though C for the sake of clarity. It is to be appreciated however, that a different weight vector may be employed for the top-level classifier and for each of the sublevel classifiers.

Referring now to Fig. 4, a system 80 depicts a more detailed information search and retrieval system in accordance with the present invention. The system 80 includes an interactive component such as a context disambiguation system (CDS) 84 for analyzing the probabilistic results from the ACS 58, controlling a dialog feedback 88 to a display or other computer system output (not shown), and rendering search results to the display. The CDS 84 includes an analytical component 90 and a presentation component 92 for directing output to the computer system. The output may include text, audio and/or other feedback relating to the search results/dialog feedback 88. As described above, the query input 70 is provided to the ACS 58 wherein a top-level results output 96 is initially directed to the analytical component 90. The top-level results 96 are the probabilistic weights determined for the category areas by the top-level classifier in the ACS 58.

The results 96 are processed by the analytical component 90 to disambiguate the

category areas based on the determined probabilities. Upon determining the most likely category areas, which is described in more detail below, the analytical component 90 drives the presentation component 92 to display these category areas *via* the dialog feedback 88. The dialog may range from sophisticated text-generation engaging the user to clarify the category area of focus, to a simpler presentation of a list containing topics that the user may desire to review, coupled with a question seeking feedback from the user that can disambiguate the focus of the list. Upon reviewing the dialog, the user then indicates one or more suitable category areas *via* a selection input 98 such as a mouse, keyboard or other input. Based upon the user's selection, the analytical component drives a system feedback 100 to indicate the category areas selected. The ACS 58 enables a sublevel classifier associated with the category area indicated by the system feedback 100 and utilizes the query input 70 to determine a sublevel result 104 indicating the most likely topics relating to the category area. The analytical component 90 may then determine which topics should be directed to the presentation component 92 based upon the probabilities contained in the results 104. These topics are then provided to the user as search results *via* the dialog feedback 88 wherein the user may then select the desired topic of interest to review. This process can be repeated for respective levels in the hierarchy, thus a sublevel loop can be iterated over a number of times.

Several analytical techniques and policies based on probabilities may be applied by the analytical component 90 to the top and sublevel results 96 and 104 in determining the dialog feedback 88 to display. These techniques may include a statistical cost benefit analysis and/or decision analysis to determine whether to inquire for more feedback 88 from the user or when to display search results. For example, based upon the spread of probabilities across the categories and topics (*e.g.*, probabilities equally weighted across many categories or heavily weighted toward a few categories), it may be determined that dialog feedback posing a question is necessary to help clarify which category the user may be interested in. If the probabilities were heavily weighted toward a single category, it may be determined to directly display topics without first requiring feedback, for example. These decisions may be base upon a rule-based policy that controls if and how dialog should be invoked based on the distribution of probabilities assigned to topics at one or more layers of the classification scheme. For example, a rule may be established

that the top five results are displayed unless the probability for the category or topic is below a predetermined probability threshold. Alternatively, an expected-utility policy may be utilized that controls if and how dialog should be invoked based on the distribution of probabilities assigned to topics at one or more layers of the classification scheme. These type of policies may be based on assigning costs associated with the display and browsing of sets of results versus the costs of the steps of inquiring more feedback. The costs may be multiplied by topic or category probabilities to determine the expected-utility of displaying a result or question.

Other decision-making techniques and/or policies provided by the analytical component can include a cost-benefit analysis that considers the cost of the dialog with the information value of the dialog. This can also include a decision analysis for determining the nature and quantity of a clarification dialog, for example. Still yet other analyses can include a computation of the value of information associated with feedback gained during a clarification dialog to guide the nature and quantity of the clarification dialog. This can include employing an expected value of information (EVI) construct that is well-known within the fields of decision theory and analysis.

Another aspect of the present invention includes providing a background monitor 110 to monitor explicit and/or implicit user responses/actions within the system 80. For example, the background monitor 110 may receive inputs from various locations within the system 80 and record those instances of direct and/or indirect activity associated with the inputs within a local data store 112. These inputs may include the query input 70, the top and sublevel results 96, 104, dialog feedback 88 and the system feedback 100, for example. Explicit monitoring refers to those actions directly initiated by the user such as the content of the query itself and subsequent selections based upon the dialog feedback 88. Implicit monitoring refers to making determinations regarding temporal aspects of the user responses. For example, these aspects may include how long a particular user pauses based upon a particular dialog. The data store 112 may then be provided to the ACCS 54 in order to provide updated classifier models to the ACS 58.

Turning now to Fig. 5, a central data store and server 115 is provided to receive input data over a network 118 from a plurality of background monitors associated with a plurality of users 1 through X. In this manner, queries and other types of monitoring may

continuously be collected and utilized to generate more accurate classifiers for the ACCS 54. Also, it may be determined from the input data that other categories and/or topics should be developed if it is determined that some queries inquire about topical information not previously considered and/or stored.

Referring now to Figs. 6 and 7, and exemplary display presentation is illustrated in accordance with the present invention. Fig. 6 depicts a ranked display of potential categories 1 through J that have been determined by the analytical component 90 described above. Selection inputs 120-123 may be provided next to each displayed category to enable the user to input feedback regarding the category area or areas of interest. The ranked displayed may also include probabilities P1 through PJ indicating the likelihood that a particular category may be desired for the user. In this example, the user may select Category 2 as the category of interest. Based upon the selection, a display presentation as depicted in Fig. 7 may be provided. In this example, potential topics ranked 1 through T associated with the users query and relating to Category 2 are then displayed. As illustrated, the topics may also include probabilities. In this example, the user may select topic 1 at reference numeral 124 to retrieve information associated with the selected topic.

Referring to Fig. 8, an alternative aspect of the present invention is depicted relating to probabilities for the layers 20. As described above, probabilities may be determined *via* classifiers. According to this aspect of the invention, these probabilities may also be predetermined and stored on a database 130. The database 130 of the present invention contains various information utilized by the analytical component 90 to process the queries and determine potential categories and topics. This information may consist of seven data types, for example: topics, synonyms, metanyms, functional words, probabilities, links, and metalinks.

The "topics" are the types of on-line assistance provided to a user. An example of a topic may be "create a new chart," which would provide information to a user of a spreadsheet program on how to create a chart. The topics are written off-line by the developers of the software product based on the types of problems that users usually have in utilizing the software product. Associated with each topic of the present invention is a "prior probability." The prior probability is the likelihood that a user would need a

particular topic in the absence of any other information. For example, the topic "create a new chart" may have a prior probability of .05, whereas the topic "edit a custom chart using a macro" might have a prior probability of .01. The value of these two prior probabilities indicates that the "edit a custom chart using a macro" topic is much less likely to be needed by a user. The prior probabilities are created by the developers of the software product based on experience and customer research.

The "synonyms" data type are words that a user may use to refer to each topic (*i.e.*, their help needs). For example, for the topic "create a new chart" the synonyms may include "new," "chart," "graph," "make," "create," and "picture." The synonyms for each topic are created off-line by the software product developers based on experience and customer research.

The "metanyms" data type is a general classification for synonyms. That is, a metanym is a higher-level conceptual grouping of synonyms. For example, the synonyms "blue," "red," "green," and "yellow" can be grouped into the metanym "color." Another example is the synonyms "chart" and "graph" can be grouped into the metanym "chart." Metanyms can also be a higher-level grouping of spelling variations, as well as different types of contractions. By utilizing metanyms, the present invention reduces the number of words that are considered when performing probability analysis. Metanyms are created off-line by the software developers based on experience and customer research.

The "functional words" data type is a grouping of common articles, possessives, demonstratives, prepositions, and other similar words. Functional words are used in definiteness analysis to determine the form of a word. For example, the word "my" in the phrase "my chart" indicates that "chart" is in the definite form, whereas the word "a" in the phrase "a chart" implies that "chart" is in the indefinite form. The list of functional words comes from the experience of the software developers, as well as customer research.

The "probabilities" data type indicates the likelihood that a user will use a particular metanym to identify a topic. For example, if a customer wants the "create a new chart" topic, there is a high probability that the user will use the word "chart" in the input. However, if the user wants the "print a document" topic, there is a much lower

probability that the user will use "chart" in the input. The probabilities are created off-line based on the experience of the software developers and customer research.

The "links" data type is a connection between a metanym and a topic. Therefore, a link is employed to indicate that a metanym may be used to refer to a topic. A "link relevance" is maintained as part of the link data type. The link relevance is an indication of the expectation that a metanym will be used in a definite form, an indefinite form, or a neutral form when a user requests the linked topic. For example, it is more likely when a user requests the "create a new chart" topic that a user will use the indefinite form as opposed to the definite form, because the chart does not exist yet. Links are created based on the experience of the software developers and customer research. The "metalinks" data type is used to link synonyms with the associated metanym. As with all other data types, metalinks are created off-line using the experience of the software developers and customer research.

Fig. 9 illustrates a methodology for providing information search and retrieval in accordance with an aspect of the present invention. While, for purposes of simplicity of explanation, the methodology is shown and described as a series of acts, it is to be understood and appreciated that the present invention is not limited by the order of acts, as some acts may, in accordance with the present invention, occur in different orders and/or concurrently with other acts from that shown and described herein. For example, those skilled in the art will understand and appreciate that a methodology could alternatively be represented as a series of interrelated states or events, such as in a state diagram. Moreover, not all illustrated acts may be required to implement a methodology in accordance with the present invention.

Referring to Fig. 9, and proceeding to 150, top-level and sublevel classifiers are constructed as described above for determining probabilities associated with a query. At 154, a query is processed in conjunction with the top-level classifier in order determine the probabilities associated with a category area. At 158, the most likely categories are determined based upon the probabilities determined in 154 and the results are provided to the user. As described above, policies may be established to determine the categories that are actually provided and/or displayed to the user. At 162, user feedback is received in the form of a category selection provided by the user relating to a desired area of interest.

At 166, a sublevel classifier associated with the category relating to the users selection in 162 is employed to find a desired topic utilizing the query input. As described above, when categories and topics are displayed, associated probabilities may also be displayed that relate to a particular topic or category. It is noted that acts 158 through 166 can be iterated for each respective level of the hierarchy.

In order to provide a context for the various aspects of the invention, Fig. 10 and the following discussion are intended to provide a brief, general description of a suitable computing environment in which the various aspects of the present invention may be implemented. While the invention has been described above in the general context of computer-executable instructions of a computer program that runs on a computer and/or computers, those skilled in the art will recognize that the invention also may be implemented in combination with other program modules. Generally, program modules include routines, programs, components, data structures, *etc.* that perform particular tasks and/or implement particular abstract data types. Moreover, those skilled in the art will appreciate that the inventive methods may be practiced with other computer system configurations, including single-processor or multiprocessor computer systems, minicomputers, mainframe computers, as well as personal computers, hand-held computing devices, microprocessor-based or programmable consumer electronics, and the like. The illustrated aspects of the invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. However, some, if not all aspects of the invention can be practiced on stand-alone computers. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

With reference to Fig. 10, an exemplary system for implementing the various aspects of the invention includes a computer 220, including a processing unit 221, a system memory 222, and a system bus 223 that couples various system components including the system memory to the processing unit 221. The processing unit 221 may be any of various commercially available processors. Dual microprocessors and other multi-processor architectures also can be used as the processing unit 221.

The system bus may be any of several types of bus structure including a memory

bus or memory controller, a peripheral bus, and a local bus using any of a variety of commercially available bus architectures. The system memory may include read only memory (ROM) 224 and random access memory (RAM) 225. A basic input/output system (BIOS), containing the basic routines that help to transfer information between elements within the computer 220, such as during start-up, is stored in ROM 224.

The computer 220 further includes a hard disk drive 227, a magnetic disk drive 228, *e.g.,* to read from or write to a removable disk 229, and an optical disk drive 230, *e.g.,* for reading from or writing to a CD-ROM disk 231 or to read from or write to other optical media. The hard disk drive 227, magnetic disk drive 228, and optical disk drive 230 are connected to the system bus 223 by a hard disk drive interface 232, a magnetic disk drive interface 233, and an optical drive interface 234, respectively. The drives and their associated computer-readable media provide nonvolatile storage of data, data structures, computer-executable instructions, etc. for the computer 220. Although the description of computer-readable media above refers to a hard disk, a removable magnetic disk and a CD, it should be appreciated by those skilled in the art that other types of media which are readable by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, and the like, may also be used in the exemplary operating environment, and further that any such media may contain computer-executable instructions for performing the methods of the present invention.

A number of program modules may be stored in the drives and RAM 225, including an operating system 235, one or more application programs 236, other program modules 237, and program data 238. The operating system 235 in the illustrated computer may be substantially any commercially available operating system.

A user may enter commands and information into the computer 220 through a keyboard 240 and a pointing device, such as a mouse 242. Other input devices (not shown) may include a microphone, a joystick, a game pad, a satellite dish, a scanner, or the like. These and other input devices are often connected to the processing unit 221 through a serial port interface 246 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, a game port or a universal serial bus (USB). A monitor 247 or other type of display device is also connected to the system bus 223 *via* an interface, such as a video adapter 248. In addition to the monitor, computers typically

include other peripheral output devices (not shown), such as speakers and printers.

The computer 220 may operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 249. The remote computer 249 may be a workstation, a server computer, a router, a peer device or other common network node, and typically includes many or all of the elements described relative to the computer 220, although only a memory storage device 250 is illustrated in Fig. 10. The logical connections depicted in Fig. 10 may include a local area network (LAN) 251 and a wide area network (WAN) 252. Such networking environments are commonplace in offices, enterprise-wide computer networks, Intranets and the Internet.

When employed in a LAN networking environment, the computer 220 may be connected to the local network 251 through a network interface or adapter 253. When utilized in a WAN networking environment, the computer 220 generally may include a modem 254, and/or is connected to a communications server on the LAN, and/or has other means for establishing communications over the wide area network 252, such as the Internet. The modem 254, which may be internal or external, may be connected to the system bus 223 *via* the serial port interface 246. In a networked environment, program modules depicted relative to the computer 220, or portions thereof, may be stored in the remote memory storage device. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be employed.

In accordance with the practices of persons skilled in the art of computer programming, the present invention has been described with reference to acts and symbolic representations of operations that are performed by a computer, such as the computer 220, unless otherwise indicated. Such acts and operations are sometimes referred to as being computer-executed. It will be appreciated that the acts and symbolically represented operations include the manipulation by the processing unit 221 of electrical signals representing data bits which causes a resulting transformation or reduction of the electrical signal representation, and the maintenance of data bits at memory locations in the memory system (including the system memory 222, hard drive 227, floppy disks 229, and CD-ROM 231) to thereby reconfigure or otherwise alter the computer system's operation, as well as other processing of signals. The memory

locations wherein such data bits are maintained are physical locations that have particular electrical, magnetic, or optical properties corresponding to the data bits.

What has been described above are preferred aspects of the present invention. It is, of course, not possible to describe every conceivable combination of components or methodologies for purposes of describing the present invention, but one of ordinary skill in the art will recognize that many further combinations and permutations of the present invention are possible. Accordingly, the present invention is intended to embrace all such alterations, modifications and variations that fall within the spirit and scope of the appended claims.